

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 04-157549

(43)Date of publication of application : 29.05.1992

(51)Int.Cl.

G06F 13/00

(21)Application number : 02-281778

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 22.10.1990

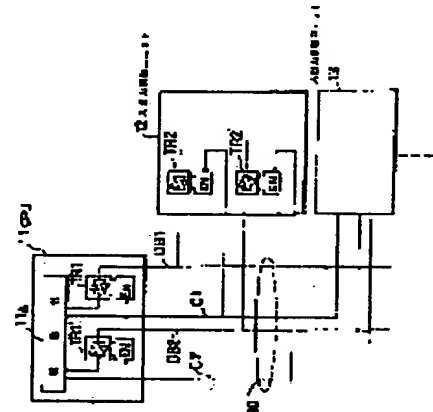
(72)Inventor : HAMADA YOICHI

## (54) BUS CONTROL SYSTEM

### (57)Abstract:

**PURPOSE:** To enable a data processor to continue to operate without any stop even if data bus trouble occurs by performing degenerative operation which uses a normal data bus in case of the data bus trouble occurrence.

**CONSTITUTION:** The data processor is a computer system consisting of a system bus 10, a CPU 11, input/output units 12, 13..., and those CPU 11 and input/output control units 12, 13... are mounted on a mount panel and connected to the system bus 10 respectively. Here, the degenerative operation which utilizes the normal data bus is performed if the data bus trouble occurs. Consequently, even if the data bus trouble occurs, the data processor can continue to operate without any stop.



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

⑩ 日本国特許庁(JP)

⑪ 特許出願公開

⑫ 公開特許公報(A) 平4-157549

⑬ Int.Cl.<sup>3</sup>  
G 06 F 13/00

識別記号 庁内整理番号  
3 0 1 M 7368-5B

⑭ 公開 平成4年(1992)5月29日

審査請求 未請求 請求項の数 1 (全6頁)

⑮ 発明の名称 バス管理方式

⑯ 特 願 平2-281778

⑰ 出 願 平2(1990)10月22日

⑱ 発 明 者 浜 田 洋 一 東京都青梅市末広町2丁目9番地 株式会社東芝青梅工場  
内

⑲ 出 願 人 株 式 会 社 東 芝 神奈川県川崎市幸区堀川町72番地

⑳ 代 理 人 弁理士 鈴 江 武 彦 外3名

明 細 書

1. 発明の名称

バス管理方式

2. 特許請求の範囲

複数の入出力制御装置およびCPUがシステムバスを介して結合されて成るデータ処理装置において、

前記複数の入出力制御装置とCPU間に設けられ、データ送受信をそれぞれ実行する複数のデータバスと、

これらデータバスの障害を各バス毎に検出する障害検出手段と、

この障害検出手段によって障害が検出された際、そのデータバスを不活性状態に制御すると共に、そのデータバス上でのデータ転送を他のデータバスに切り替えるバス制御手段とを具備し、

障害発生時に正常なデータバスを利用した縮退運転を行うことを特徴とするバス管理方式。

3. 発明の詳細な説明

[発明の目的]

(産業上の利用分野)

この発明はバス管理方式に関し、特に複数の入出力制御装置およびCPUがシステムバスを介して結合されて成るデータ処理装置におけるデータバスの管理方式に関する。

(従来の技術)

一般に、データ処理装置においては、複数の入出力制御装置およびCPUがシステムバスを介して結合されているが、そのシステムバス、特にデータバスで障害が発生するとシステムの運転停止となる場合が多い。

データバスの障害は、データバスを伝達されるデータの論理が固定的に定められてしまう等のデータ誤りを招くものであり、このような障害は、例えば、各入出力装置においてそのデータバスに接続されているトランシーバ回路またはその前段回路等の故障を原因として引き起こされる。

すなわち、トランシーバ回路等に故障が発生す

ると、そのトランシーバ回路を備えている入出力制御装置は所望のデータを送信することができなくなり、また他の入出力制御装置やCPUもデータバスを介して転送される誤ったデータを受信することになるので、異常動作を始める。したがって、データバス障害、つまりデータバスに接続されているトランシーバ回路等の故障は、データ処理装置において致命的な障害であり、このような障害が発生した場合には、即座にシステムの運転停止となる。

このように、従来では、データバス障害が発生すると、データ処理装置全体に影響が及ぼされるので、データ処理装置の稼働を停止しなければならない欠点があった。

(発明が解決しようとする課題)

従来では、データバス障害が発生すると、データ処理装置全体に影響が及ぼされるので、データ処理装置の稼働を停止しなければならない欠点があった。

この発明はこのような点に鑑みてなされたも

ば高インピーダンス状態等の不活性状態に制御されるので、そのデータバス障害がどの入出力制御装置に起因するものであっても、誤ったデータの転送がなされることを防止でき、他の入出力制御装置やCPUには障害による影響は及ぼされない。また、このような障害時には、正常なデータバスを利用した縮退運転に切り替えられるので、データ処理装置の稼働を停止すること無く、運転を継続することができる。

(実施例)

以下、図面を参照してこの発明の実施例を説明する。

第1図にはこの発明の一実施例に係わるデータ処理装置の構成が示されている。

このデータ処理装置は、システムバス10、CPU11、および入出力制御ユニット12、13、…によって構成されるコンピュータシステムであり、これらCPU11、および入出力制御ユニット12、13は図示しない装着パネルに装着されて、システムバス10にそれぞれ接続されている。

ので、データバス障害が発生しても、運転を停止すること無くデータ処理装置の動作を継続して実行することができるバス管理方式を提供することを目的とする。

(発明の構成)

(課題を解決するための手段および作用)

この発明によるバス管理方式は、複数の入出力制御装置およびCPUがシステムバスを介して結合されて成るデータ処理装置において、前記複数の入出力制御装置とCPU間に設けられ、データ送受信をそれぞれ実行する複数のデータバスと、これらデータバスの障害を各バス毎に検出する障害検出手段と、この障害検出手段によって障害が検出された際、そのデータバスを不活性状態に制御すると共に、そのデータバス上でのデータ転送を他のデータバスに切り替えるバス制御手段とを具備し、障害発生時に正常なデータバスを利用した縮退運転を行うことを特徴とする。

このバス管理方式においては、複数のデータバスのうちで障害が発生したデータバスは、例え

システムバス10は、通常のように、データバス、アドレスバス、および各種コントロール線から構成されるものであるが、ここでは、データバスとして2つの分離されたデータバスDB1、DB2が設けられている。データバスDB1、DB2は、例えばそれぞれ8ビット構成である。また、コントロール線群の一部として、データバスDB1を不活性制御するための制御線C1と、データバスDB2を不活性制御するための制御線C2が設けられている。

データバスDB1、DB2は、CPU11、入出力制御ユニット12、13の各々における対応する双方向ドライバ(トランシーバ)に接続されている。制御線C1は、CPU11、入出力制御ユニット12、13の各々におけるデータバスDB1が接続された双方向ドライバのイネーブル端子ENに接続されている。

CPU11は、このデータ処理装置全体の制御および各種演算を実行するものであり、制御部11a、第1の双方向ドライバTR1、第2の双方向ドラ

イバT R1'等をプリント回路ボード上に配設して構成されている。

第1の双方向ドライバT R1は第1のデータバスD B1に接続され、そのデータバスD B1に対して例えば8ビットのデータの送受信を行う。双方向ドライバT R1にはイネーブル端子E Nが設けられており、このイネーブル端子E Nには制御線C1が接続されている。すなわち、第1の双方向ドライバT R1は、イネーブル端子E NでLレベルの制御信号を制御線C1から受信した時には活性状態に制御されて第1のデータバスD B1とデータを授受し、Hレベルの制御信号を制御線C1から受信した時には不活性状態に制御されて第1のデータバスD B1を高インピーダンス(H i-Z)状態に設定する。

同様に、第2の双方向ドライバT R1'は第2のデータバスD B2に接続され、そのデータバスD B2に対して例えば8ビットのデータの送受信を行う。双方向ドライバT R1'にはイネーブル端子E Nが設けられており、このイネーブル端子

で転送する。

入出力制御ユニット12、13は、C P U11の制御の下に図示しない外部の入出力装置を制御するためのものであり、C P U11と同様に、第1のデータバスD B1に接続される第1の双方向ドライバT R2と、第2のデータバスD B2に接続される第2の双方向ドライバT R2'を備えている。

すなわち、入出力制御ユニット12における第1の双方向ドライバT R2は第1のデータバスD B1に接続され、そのデータバスD B1に対して例えば8ビットのデータの送受信を行う。双方向ドライバT R2にはイネーブル端子E Nが設けられており、このイネーブル端子E Nには制御線C1が接続されている。第1の双方向ドライバT R2は、イネーブル端子E NでLレベルの制御信号を制御線C1から受信した時には活性状態に制御されて第1のデータバスD B1とデータを授受し、Hレベルの制御信号を制御線C1から受信した時には不活性状態に制御されて第1のデータバスD B1を高インピーダンス(H i-Z)状態

E Nには制御線C2が接続されている。すなわち、第2の双方向ドライバT R1'は、イネーブル端子E NでLレベルの制御信号を制御線C2から受信した時には活性状態に制御されて第2のデータバスD B2とデータを授受し、Hレベルの制御信号を制御線C2から受信した時には不活性状態に制御されて第2のデータバスD B2を高インピーダンス(H i-Z)状態に設定する。

制御部11aは、マイクロコンピュータ等から構成されるものであり、データ転送に関する各種制御や、データバスD B1、D B2に異常監視等を行う。例えば、データバスD B1、D B2の双方を利用した16ビットのデータ転送の実行中において、データバスD B1の障害を検出した場合には、制御線C1をLレベルからHレベルに変化させて、そのデータバスD B1に繋がる双方向ドライバを不活性状態に設定する。そして、データバスD B1上の8ビット単位のデータ転送をデータバスD B2に肩代わりさせるために、データバスD B2を用いてデータを時分的に8ビット単位

に設定する。

同様に、入出力制御ユニット12の第2の双方向ドライバT R2'は第2のデータバスD B2に接続され、そのデータバスD B2に対して例えば8ビットのデータの送受信を行う。双方向ドライバT R2'にはイネーブル端子E Nが設けられており、このイネーブル端子E Nには制御線C2が接続されている。すなわち、第2の双方向ドライバT R2'は、イネーブル端子E NでLレベルの制御信号を制御線C2から受信した時には活性状態に制御されて第2のデータバスD B2とデータを授受し、Hレベルの制御信号を制御線C2から受信した時には不活性状態に制御されて第2のデータバスD B2を高インピーダンス(H i-Z)状態に設定する。

第2図には、制御部11aの具体的な構成の一例が示されている。

この制御部11aは、16ビットのレジスタR1、第1および第2のセレクトS1、S2、および切替え制御回路S Wを備えている。

レジスタR1には、CPU11がデータバスDB1、DB2を介して送信すべきデータが保持されており、レジスタR1の左側の8ビットのデータ(L)は、セレクトAS1、S2に共通に供給されている。また、レジスタR1の右側の8ビットのデータ(R)も、セレクトAS1、S2に共通に供給されている。

セレクトAS1は、第1の選択信号SEL1を受信した時には右側の8ビットのデータ(R)を選択し、第2の選択信号SEL2を受信した時には左側の8ビットのデータ(L)を選択する。

セレクトAS2は、第1の選択信号SEL1を受信した時には左側の8ビットのデータ(L)を選択し、第2の選択信号SEL2を受信した時には右側の8ビットのデータ(R)を選択する。

切替え制御回路SWは、データバスDB1、DB2の異常監視、およびセレクトAS1、S2の制御を行うものであり、通常は、第1の選択信号SEL1を順次出力して、右側の8ビットのデータ(R)を双方向ドライバTR1に送信し、左側

の8ビットのデータ(L)を双方向ドライバTR1'に送信する。また、例えば、データバスDB1の障害を検出した場合には、制御線C1をHレベルにしてデータバスDB1を高インピーダンスに設定すると共に、第1の選択信号SEL1と第2の選択信号SEL2を交互に出力して、右側の8ビットのデータ(R)と左側の8ビットのデータ(L)を交互に双方向ドライバTR1'に送信する。

次に、第3図のタイミングチャートおよび第4図のフローチャートを参照して、データバスの障害監視動作とデータ転送の切替え動作を説明する。

CPU11は、データバスDB1、DB2を常時監視しているが、この監視処理においては、まず双方向ドライバTR1、TR1'を介して対応するバスからデータが受信される(ステップS1)。

次いで、その受信データを良く知られたパリティチェック等の手段を用いてチェックし、バスエラー(障害)が発生しているか否かを判断する(ステップS2)。

例えば、第1のデータバスDB1におけるバスエラーが検出された場合には、その検出の度にデータバスDB1に対応するエラーフラグをセットしてそのフラグ(FLG)の値を増分する(ステップS3)。

そして、そのフラグ(FLG)の値が1以下であるか否かが一定期間毎に判断され(ステップS4)、1以下である場合には、障害発生検出の正確性を維持するために、再度データ受信を行う。このデータ受信は、例えば、CPU11が同一のデータを再び送信して、その応答状態を再度検出するといった再試行処理を意味するものである。

一方、フラグ(FLG)の値が2以上である場合には、バス障害が間違いなく発生していると認識され、その障害が検出されたデータバスの切り離し処理が実行される(ステップS5)。

この切り離し処理では、例えばデータバスDB1の障害(例えば、第3図に示されているようにデータバスDB1がLレベルに固定される障害)が検出された場合においては、制御線C1が

LレベルからHレベルに切り替えられ、これによってデータバスDB1は高インピーダンスの不活性状態となる。

次いで、第2図の切替え制御回路SWは、データバスDB1のデータ転送をデータバスDB2で隔代わりするために、第1の選択信号SEL1と第2の選択信号SEL2を交互に出力して、右側の8ビットのデータ(R)と左側の8ビットのデータ(L)を交互に双方向ドライバTR1'に送信して、データ(R)とデータ(L)をデータバスDB2に時分割的に送信する。

そして、入出力制御ユニット12、13は、データバスDB1が高インピーダンス状態に設定されたこと、あるいは制御線C1がHレベルに設定された事によって、データバスDB1が使用不可能であることを認識し、データバスDB2を利用した縮退運転が行われる。

以上のように、この実施例においては、障害が発生したデータバスは、高インピーダンス状態に設定制御されるので、そのデータバス障害がどの

入出力制御ユニットの双方向ドライバに起因するものであっても、誤ったデータの転送がなされることを防止でき、他の入出力制御ユニットやCPUにはその障害による影響は及ぼされない。また、このような障害時には、正常なデータバスを利用した縮退運転に切り替えられるので、データ処理装置の稼働を停止すること無く、運転を継続することができる。

尚、ここでは、データバスの障害監視機構をCPUにのみ設けた場合を説明したが、第2図と同様の回路を入出力制御ユニット12, 13に設けても良いことはもちろんである。

#### 【発明の効果】

以上のように、この発明によれば、データバス障害が発生しても、運転を停止すること無くデータ処理装置の動作を継続して実行することが可能となる。

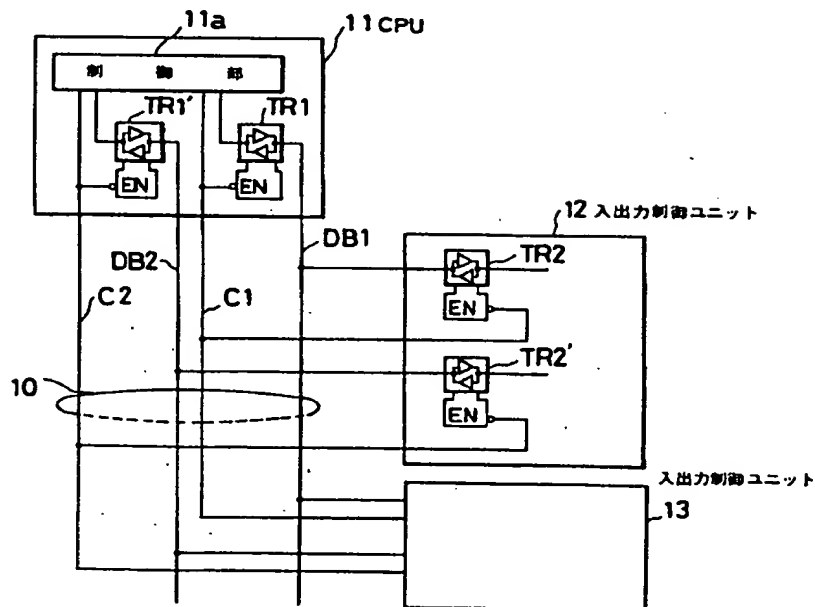
#### 4. 図面の簡単な説明

第1図はこの発明の一実施例に係わるバス管理方式を用いたデータ処理装置の構成を示すブロッ

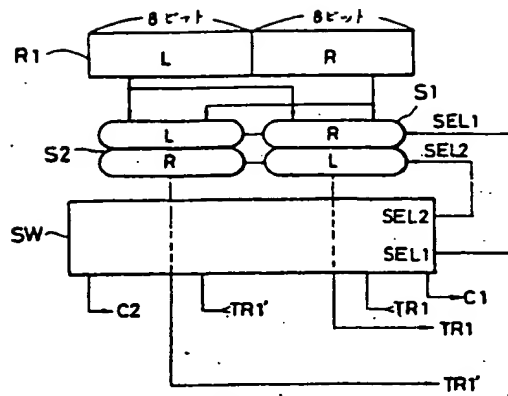
ク図、第2図は第1図のデータ処理装置に設けられたCPUの具体的な構成の一例を示すブロック図、第3図および第4図はそれぞれ第1図のデータ処理装置によるバス管理動作を説明するための図である。

11…CPU、12, 13…入出力制御ユニット、DB1, DB2…データバス、TR1, TR1'…双方向ドライバ。

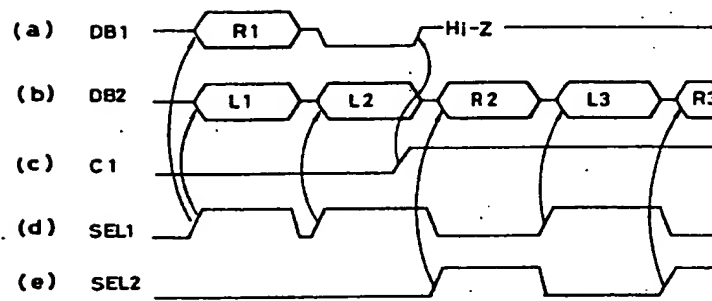
出願人代理人 弁理士 鈴江武彦



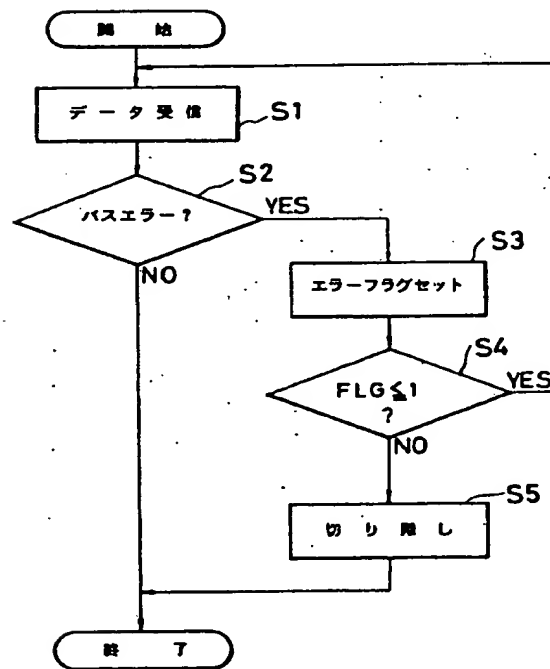
第1図



第 2 図



第 3 図



第 4 図

## SPECIFICATION

### 1. TITLE OF THE INVENTION

Bus Control System

### 2. CLAIM

In a data processor wherein a plurality of input/output control units and a CPU are coupled to each other via system buses, a bus control system comprising:

a plurality of data buses which are arranged between said plurality of input/output control units and said CPU to independently execute transmission and receiving of data;

fault detecting means for detecting a fault that may occur in those data buses for each bus; and

bus control means which, upon the detection of a fault by said fault detection means, controls the data bus concerned in an inactive status and also switches data transfer by said data bus concerned over to data transfer by another data bus;

wherein, when a data bus fault occurs, a degenerative operation utilizing a normal bus is performed.

### 3. DETAILED DESCRIPTION OF THE INVENTION

[Objects of the Invention]



(Utilization Field in Industry)

The present invention relates to a bus control system, and more particularly to a data bus control system in a data processor consisting of a plurality of input/output control units and a CPU that are coupled via system buses.

(Prior Art)

With data processors in general, a plurality of input/output units and a CPU are coupled to each other via system buses, and an occurrence of a fault in a system bus, more specifically in a data bus, may likely result in an operational stop of the system.

A data bus fault will lead to data errors including a case where a data logic to be delivered through the data bus is determined to be almost unchangeable. Such fault will be triggered, for example, in respective input/output units, as a result of a failure in a circuit such as a transceiver circuit which is connected to the data bus or a pre-stage circuit of the transceiver circuit.

More specifically, if a fault occurs in the transceiver circuit, for example, an input/output control unit which incorporates the transceiver circuit becomes unable to transmit desired data. As a result, other input/output control units and the CPU will present abnormal operations since they receive wrong data that are

transferred via the data bus. Therefore, a data bus fault, or a failure in the transceiver circuit that is connected to the data bus constitutes a fatal fault in the data processor, and the occurrence of such fault will immediately result in an operational stop of the system.

As described in the above, conventional data processors have a disadvantage that the operation of a data processor has to be stopped in case of the occurrence of a data bus fault, since the fault affects the entire data processor.

(Problem to be Solved by the Invention)

Conventional data processors have a disadvantage that the operation of data processor has to be stopped in case of the occurrence of a data bus fault, since a data bus fault affects the entire data processor.

It is therefore an object of the present invention to overcome the above drawback of the prior art and to provide a bus control system which can continuously execute data processor operations without stopping the operation even if a data bus fault occurs.

[Constitution of the Invention]

(Means and Operations to Solve the Problem)

According to the present invention, in a data processor wherein a plurality of input/output control units and a CPU are coupled to each other via system buses, a bus

control system comprises: a plurality of data buses which are arranged between said plurality of input/output control units and said CPU to independently execute transmission and receiving of data; fault detecting means for detecting a fault that may occur in those data buses for each bus; and bus control means which, upon the detection of a fault by said fault detection means, controls the data bus concerned in an inactive status and also switches data transfer by said data bus concerned over to data transfer by another data bus; wherein, when a data bus fault occurs, a degenerative operation utilizing a normal bus is performed.

With the bus control system of the present invention, among a plurality of data buses, a specific data bus having a fault is controlled in an inactive status of a high impedance, for example, and therefore, the transfer of wrong data can be prevented even if the data bus fault is attributed to whichever input/output control unit. In addition, other input/output control units and the CPU will not be affected by the fault. Furthermore, if any fault occurs, the system can be switched over to a degenerative operation using a normal data bus, and the operation can be continued without stopping the operation of the data processor.

(Embodiment)

Preferred embodiments of the present invention will now be described in detail with reference to the accompanying drawings.

Fig. 1 is a drawing illustrating a configuration of a data processor in accordance with a preferred embodiment of the present invention.

The data processor is a computer system including system buses 10, a CPU 11, and input/output control units 12, 13 ..., and those CPU 11 and input/output control units 12, 13 ... are mounted on a mount panel that is not shown on the drawing and connected to the system bus 10 respectively.

The system buses 10 includes, as usual systems, a data bus, an address bus and various control lines, but as illustrated in the drawing, two separate data buses of DB1 and DB2 are provided for the present invention. The data buses DB1 and DB2 are respectively of an 8-bit configuration. In addition, as part of a group of control lines, a control line C1 which is used to give an inactive control of the data bus DB1, and a control line C2 which is used to give an inactive control of the data bus DB2 are provided.

The data buses DB1 and DB2 are connected to a two-way driver (transceiver) which respectively corresponds to the CPU 11 and input/output control units 12 and 13. The control line C1 is connected to an enable terminal EN of

the two-way driver, to which the data bus DB1 is connected, of the CPU 11 and the input/output control units 12 and 13.

The CPU 11 controls the entire data processor and executes various calculations, and is so configured to incorporate a controller 11a, a first two-way driver TR1 and a second two-way driver TR1' on a printed circuit board.

The first two-way driver TR1 is connected to a first data bus DB1 and executes the transmission and the receiving of, for example, 8-bit data for the data bus DB1. The two-way driver TR1 is provided with the enable terminal EN, and the control line C1 is connected to the terminal EN. More specifically, when an L-level control signal is received from the control line C1 at the enable terminal EN, the first two-way driver TR1 is controlled to be an active status to transmit and receive data for the first data bus DB1. While, when a H-level control signal is received from the control line C1, the driver TR1 is controlled to be an inactive status to set the first data bus DB1 in the high-impedance (Hi-Z) status.

Likewise, the second two-way driver TR1' is connected to the second data bus DB2 to execute the transmission and the receiving of, for example, 8-bit data for the data bus DB2. The two-way driver TR1' is provided with another enable terminal EN, and the control line C2 is connected to the terminal EN. More specifically, when an

L-level control signal is received from the control line C2 at the enable terminal EN, the first two-way driver TR1' is controlled to be an active status to transmit and receive data for the second data bus DB2. While, when a H-level control signal is received from the control line C2, the driver TR1' is controlled to be an inactive status to set the second data bus DB2 in the high-impedance (Hi-Z) status.

The controller 11a is constituted of a micro-computer and other devices and executes tasks including various controls regarding the data transfer, and the monitoring of faults on the data buses DB1 and DB2. For example, when a 16-bit data transfer is being executed by using both of the data buses DB1 and DB2, if a fault is detected on the data bus DB1, the controller 11a changes the control line C1 to the level H from the level L and set the two-way driver that is connected to the data bus DB1 in the inactive status. Thereafter, in order to make the data bus DB2 take over the transfer of 8-bit data on the data bus DB1, the controller 11a transfers the data in an 8-bit unit in a time-sharing manner by using the data bus DB2.

The input/output control units 12 and 13 are provided to control, under the control of the CPU 11, external input/output units that are not shown on the drawing, and comprises, as is the case with the CPU 11, a first two-way driver TR2 which is connected to the first

data bus DB1 and a second two-way driver TR2' which is connected to the second data bus DB2.

More specifically, the first two-way driver TR2 of the input/output control unit 12 is connected to the first data bus DB1, and executes the transmission and the receiving of, for example, 8-bit data for the data bus DB1. The two-way driver TR2 is provided with an enable terminal EN, and the control line C1 is connected to the terminal EN. When an L-level control signal is received from the control line C1 at the enable terminal EN, the first two-way driver TR2 is controlled to be an active status to transmit and receive data for the first data bus DB1. While, when a H-level control signal is received from the control line C1, the driver TR2 is controlled to be an inactive status to set the first data bus DB1 in the high-impedance (Hi-Z) status.

Likewise, the second two-way driver TR2' of the input/output control unit 12 is connected to the second data bus DB2 to execute the transmission and the receiving of, for example, 8-bit data for the data bus DB2. The two-way driver TR2' is provided with another enable terminal EN, and the control line C2 is connected to the terminal EN. More specifically, when an L-level control signal is received from the control line C2 at the enable terminal EN, the second two-way driver TR2' is controlled to be an

active status to transmit and receive data for the second data bus DB2. While, when a H-level control signal is received from the control line C2, the driver TR2' is controlled to be an inactive status to set the second data bus DB2 in the high-impedance (Hi-Z) status.

Fig. 2 is a drawing illustrating an example of specific configuration of the controller 11a.

The controller 11a includes a 16-bit register R1, a first and a second selectors S1 and S2, and a switching control circuit SW.

The register R1 retains data which are to be transmitted by a CPU 11 via a data bus DB1 and a data bus DB2, and 8-bit data (L) located at the left side of the register R1 is supplied commonly to the selectors S1 and S2. Also, 8-bit data (R) located at the right side of the register R1 is supplied commonly to the selectors S1 and S2.

The selector S1 is designed to select the right-side 8-bit data (R) when the selector receives a first selection signal SEL1, and to select the left-side 8-bit data (L) when the selector receives a second selection signal SEL2.

On the other hand, the selector S2 is designed to select the left-side 8-bit data (L) when the selector receives a first selection signal SEL1, and to select the right-side 8-bit data (R) when the selector receives a second selection signal SEL2.



The switching control circuit SW is provided to monitor errors on the data buses DB1 and DB2 and to control the selectors S1 and S2. Usually, the circuit sequentially outputs the first selection signal SEL1 to transmit the right-side 8-bit data (R) to the two-way driver TR1, and transmit the left-side 8-bit data (L) to the two-way driver TR1'. Furthermore, if the circuit has detected any fault on the data bus DB1, for example, the circuit sets the control line C1 to the level H and sets the data bus DB1 to the high impedance status. Also, the circuit alternatively outputs the first selection signal SEL1 and the second selection signal SEL2 to alternatively transmits the right-side 8-bit data (R) and the left-side 8-bit data (L) to the two-way driver TR1'.

Next, referring to a timing chart of Fig. 3 and a flow chart of Fig. 4, operations for data bus fault monitoring and data transfer switching will be described.

The CPU 11 always monitors the data buses DB1 and DB2, and in this monitoring process, data are first received from a corresponding bus via the two-way drivers TR1 and TR1' (Step S1).

Thereafter, the received data will be checked by means of a well-known parity check method and so forth for the judgment if any parity errors have occurred (Step S2).

For example, if a bus error is detected on the first

data bus DB1, an error flag corresponding to the data bus DB1 will be set whenever such error is detected, to increment the value of the flag (FLG) concerned (Step S3).

Then, a judgment is made for every specified period of time as to if the FLG value is 1 or below (Step S4), and if the value is 1 or below, the data reception will be performed again to maintain the detection accuracy of fault occurrence. The data reception implies a re-attempting process wherein the CPU 11 transmits the same data again and detects the responding status again.

On the other hand, if the FLG value is 2 or over, the CPU 11 acknowledges that a bus error has occurred without doubt and executes a process to separate the data bus on which the error has been detected (Step S5).

In this separation process, if an error (for example, an error that the data bus DB1 is fixed at the level L as illustrated on Fig. 3) has been detected on the data bus DB1, for example, the control line C1 will be switched over to the level H from the level L, whereby the data bus DB1 is put in the inactive status of high impedance.

Then, in order to make the data bus DB2 take over the data transfer on the data bus DB1, the switching control circuit SW shown in Fig. 2 alternatively outputs the first selection signal SEL1 and the second selection signal SEL2 to alternatively transmit the right-side 8-bit

data (R) and the left-side 8-bit data (L) to the two-way driver TR1', and also transmit the data (R) and the data (L) to the data bus DB2 in a time-sharing manner.

Thereafter, from the fact that the data bus DB1 has been set to the high impedance status or the control line C1 has been set to the level H, the input/output control units 12 and 13 acknowledge that the data bus DB1 cannot be used, whereby a degenerative operation using the data bus DB2 is executed.

With the present preferred embodiments, as stated in the above, since the data bus on which a fault has occurred is set and controlled in the high-impedance status, the transfer of wrong data can be prevented even if the data bus fault is attributed to whichever two-way driver of the input/output control unit, and other input/output control units and the CUP will not be affected by the fault. In addition, if such a fault has occurred, the system can be switched over to a degenerative operation using a normal data bus, and the operation can be continued without stopping the operation of the data processor.

Although the description has been made in the above on a case where a fault monitoring mechanism for data buses is arranged only in the CPU 11, it is of course possible to arrange the mechanism, which is similar to the circuit of Fig. 2, in the input/output control units 12 and 13.

#### [Effect of the Invention]

As described in the above, according to the present invention, it is possible to continuously perform the operation of a data processor without stopping the operation even if any data bus fault has occurred.

#### 4. Brief Description of the Drawings

Fig. 1 is a block diagram showing a configuration of a data processor which employs a bus control system according to a preferred embodiment of the present invention;

Fig. 2 is a block diagram showing an example of a specific configuration of the CPU incorporated in a data processor of Fig. 1; and

Figs. 3 and 4 are respectively drawings to describe the bus control operation of the data processor of Fig. 1

11 ... CPU

12 , 13 ... input/output control unit

DB1, DB2 ... data bus

TR1, TR2 ... two-way driver

Fig. 1

11a    Controller  
12     Input/Output Control Unit  
13     Input/Output Control Unit

Fig. 2

L       8 bits  
R       8 bits

Fig. 4

          START  
S1     Data Reception  
S2     Bus Error?  
S3     Error Flag Set  
S5     Separation  
          END